16 MAY 1997 10:02:55 U.S. Patent & Trademark Office P0001

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                       *
*            Welcome to MESSENGER (APS Text) at USPTO                   *
*                                                                       *
*                                                                       *
*     The USPTO production files are current through:                   *
*     13 May 1997 for U.S. Patent Text Data.                            *
*     13 May 1997 for U.S. Current Classification data.                 *
*     13 May 1997 for U.S. Patent Image Data.                           *
*                                                                       *
*        * PLEASE USE 305-9000 FOR NEW TELEPHONE NUMBER *               *
*-***********************************************************          *
*     A new set of Messenger enhancements is available,                 *
*     effective December 9, 1996.  The new FOCUS command                *
*     allows you to display your search results in relevance-           *
*     ranked order.  Enter FOCUS and the L-number you want              *
*     to view (e.g., FOCUS L5), and then display the resulting          *
*     L-number (e.g., D L6 1-12).                                       *
*     See also HELP FOCUS.                                              *
*************************************************************           *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*  DISCLAIMER:                                                          *
*  Neither the United States Government, nor any agency                 *
*  thereof, nor any of their contractors, subcontractors or             *
*  employees make any warranty, expressed or implied,                   *
*  including any warranty of marketability of fitness for a             *
*  particular purpose; nor assumes any legal liability or               *
*  responsibility for any party's use, or the results of                *
*  such, of the data.                                                   *16 MAY 1997 1
0:02:55          U.S. Patent & Trademark Office              P0002
*                                                                       *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                 Help Desk --> 703-305-9000                            *
*                                                                       *
*     The Help Desk is staffed for APS support 7 days/week.             *
*       Monday through Friday:       6:30am - 9:00pm                  *
*       Saturday, Sunday, Holidays: 8:30am - 5:00 pm                  *
*                                                                       *
*     The Help Desk staff at this number will handle all APS            *
*     related questions.                                                *
*                                                                       *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                                                                       *
*     >>>>>>>>>>>>> NEW SUNDAY HOURS !!! <<<<<<<<<<<<<                  *
*                                                                       *
*     The APS is available:                                             *
*             6:30am - 9:00pm Monday through Friday                     *
*             7:30am - 5:00pm Saturday, Sunday, Holidays                *
*                                                                       *
*       APS is unavailable Thanksgiving Day, Christmas Day,             *
*       and New Year's Day.                                             *
*                                                                       *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
FILE 'USPAT' ENTERED AT 10:02:55 ON 16 MAY 1997

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                W E L C O M E   T O   T H E                            *
*          U. S.   P A T E N T   T E X T   F I L E                      *
```

JEAN R. HOMERE Fri May 16 11:24:07 EDT 1997

* *

16 MAY 1997 10:02:55 U.S. Patent & Trademark Office P0003

```
=> s delta stream#
        146570 DELTA
        217085 STREAM#
L1           4 DELTA STREAM#
                 (DELTA(W)STREAM#)

=> d bro
:1

1.  4,318,326, Mar. 9, 1982, Plural manual organ having transposer; Stephen
L. Howell, 84/685, 445, 655, 657, 682; 984/332, 338, 347, DIG.1 [IMAGE
AVAILABLE]
:2

2.  4,057,797, Nov. 8, 1977, All digital delta to PCM converter; Adam A.
Jorgensen, 341/77; 375/242, 247 [IMAGE AVAILABLE]
:3

3.  3,956,700, May 11, 1976, Two-feedback-path delta modulation system with
circuits for reducing pulse width modulation; Joseph Henry Condon, 375/252;
341/143 [IMAGE AVAILABLE]
:end

=> s 5479654/pn
L2           1 5479654/PN

=> d fro

US PAT NO:      5,479,654 [IMAGE AVAILABLE]          L2: 1 of 1
DATE ISSUED:    Dec. 26, 1995
TITLE:          Apparatus and method for reconstructing a file from a
                  difference signature and an original file

INVENTOR:       Mark Squibb, Kingston, NY
ASSIGNEE:       Squibb Data Systems, Inc., Kingston, NY (U.S. corp.)
APPL-NO:        08/039,702
DATE FILED:     Mar. 30, 1993
REL-US-DATA:    Continuation of Ser. No. 515,164, Apr. 26, 1990, abandoned.
INT-CL:         [6] G06F 15/401
US-CL-ISSUED:   395/600; 364/DIG.2, 962.1, 962.2, 963.3, 419.19; 362/282.1;
                  395/161
US-CL-CURRENT:  395/617; 364/282.1, 962.1, 962.2, 963.3, DIG.1, DIG.2
SEARCH-FLD:     395/600, 144; 341/51; 364/955.3, 955.5, 956.1, 962.1, 966,
                  265.2, 260.81, 260.7

REF-CITED:
                              U.S. PATENT DOCUMENTS
     3,711,863    1/1973  Bloom                    444/1
     3,715,734    2/1973  Fajans                   365/127
     4,491,934    1/1985  Heinz                    364/900
     4,558,302   12/1985  Welch                    340/347DD
     4,641,274    2/1987  Swank                    364/900
     4,807,182    2/1989  Queen                    364/900
     4,881,075   11/1989  Weng                     341/87
     5,051,947    9/1991  Messenger et al.         364/900
```

OTHER PUBLICATIONS

Sarwate, Computation of Cyclic Redundancy Checks Via Table Look-Up, Communications of the ACM, Aug. 1988 V31 No. 8 p. 1008(6).
Ramabadran et al., A Tutorial On CRC Computations, IEEE Micro, Aug. 1988, V8 Issue 4.
2 information sheets on diff(C) function in Unix.

ART-UNIT: 236
PRIM-EXMR: Kevin A. Kriess
ASST-EXMR: Lucien U. Toplu
LEGAL-REP: Rosen, Dainow & Jacobs

ABSTRACT:
Invention maintains duplicate files in safe places. A SCAN computer program creates a TOKEN Table of an earlier file. The TOKEN Table reflects the indices of successive segments of the file and the exclusive-or (XR) and Cyclic redundancy check (CRC) products of the characters in each segment. An updated file is compared to the earlier file by comparing the XR and CRC products of segments in the updated file to the XR and CRC products in the TOKEN Table. On detecting matching products for identical segments, the next segments are compared. On mismatch, the segment (window) for the updated file is bumped one character and new XR and CRC products generated and compared. The indices of the TOKEN Table and the offsets from the start of the file of the first characters of the updated file matching segments are set forth in a Match Table. Next the updated file is scrolled through for the non-matching information determined by acting on the indices and offsets of the Match Table to form the TRANSITION Table which is the Match Table and the updated file non-matching information. The TRANSITION Table contains the delta information which may be sent to another location having a copy of the earlier file thereat: the whole updated file need not be sent there. A reconstruction program at the location looks at the TRANSITION Table to determine where to get the characters for the copy of the updated file it is creating.
28 Claims, 17 Drawing Figures

```
=> s 5479654/uref
L3            0 5479654/UREF

=> s ((sequen? or stream# or chain# or segment# or frame#) and (character# or b
yte or data or bit#) and (match? or compar? or differ? or delta))/ab
          49524 SEQUEN?/AB
          32245 STREAM#/AB
          30377 CHAIN#/AB
          24381 SEGMENT#/AB
          87801 FRAME#/AB
          11238 CHARACTER#/AB
           1213 BYTE/AB
          81409 DATA/AB
          25939 BIT#/AB
          17099 MATCH?/AB
          74391 COMPAR?/AB
         155553 DIFFER?/AB
           4703 DELTA/AB
L4         5348 ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (CHA
                RACTER# OR BYTE OR DATA OR BIT#) AND (MATCH? OR COMPAR? OR DIF
                FER? OR DELTA))/AB

=> s ((sequen? or stream# or chain# or segment# or frame#) and (character# or b
yte or data or bit#) and (match? or compar? or differ? or delta))/clm
         111628 SEQUEN?/CLM
          53721 STREAM#/CLM
```

```
          61276 CHAIN#/CLM
          58749 SEGMENT#/CLM
         151270 FRAME#/CLM
          17785 CHARACTER#/CLM
           2694 BYTE/CLM
         116576 DATA/CLM
          44872 BIT#/CLM
             35575 MATCH?/CLM
         152646 COMPAR?/CLM
         345078 DIFFER?/CLM
          13737 DELTA/CLM
L5        27268 ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (CHA
                RACTER# OR BYTE OR DATA OR BIT#) AND (MATCH? OR COMPAR? OR DIF
                FER? OR DELTA))/CLM

=> s (version? and (revis? or updat? or manag?))/ti
             77 VERSION?/TI
             58 REVIS?/TI
            430 UPDAT?/TI
           2242 MANAG?/TI
L6           20 (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/TI

=> s (version? and (revis? or updat? or manag?))/ab
           3036 VERSION?/AB
            399 REVIS?/AB
           4579 UPDAT?/AB
           4634 MANAG?/AB
L7          127 (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/AB

=> s (version? and (revis? or updat? or manag?))/clm
           2993 VERSION?/CLM
            879 REVIS?/CLM
          10062 UPDAT?/CLM
           5504 MANAG?/CLM
L8          320 (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/CLM

=> s l5 and l8
L9           72 L5 AND L8

=> s l4 and l7
L10           3 L4 AND L7

=> d bro
:1

1.  5,469,431, Nov. 21, 1995, Method of and apparatus for channel mapping
with relative service identification; James W. Wendorf, et al., 370/254, 431,
478; 455/4.1 [IMAGE AVAILABLE]
:ab

US PAT NO:      5,469,431 [IMAGE AVAILABLE]              L10: 1 of 3

ABSTRACT:
The availability and location of a multiplicity of multiplexed digital
information streams, transmitted over one of a multiplicity of channels,
are identified by transmitting a Global Channel Map which includes a
version number, one Band Map Table, one Channel Map Table, and at least as
many Service Map tables as there are combinations of identical service types
transmitted over different channels. The Band Map Table preferably contains
an entry for each frequency band used by the system, and a Band Map Table
```

version number. A Global Channel Map **Updates** file is transmitted in each band. Where a plurality of channels contain similar-types of services, one Service Map Table contains relative service identification numbers for those service types. The entries for relevant channels in the Channel Map Table contain base service identification **data** which are combined with the relative service identification number to locate a particular service transmitted over that channel.
:2

2. 5,317,729, May 31, 1994, Method for the storage of multi-versioned data with retrieval based on searched query; Sujan K. Mukherjee, et al., 395/603; 364/282.1, 283.1, DIG.1; 395/619 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,317,729 [IMAGE AVAILABLE] L10: 2 of 3

ABSTRACT:
A method for storage and retrieval of both time-oriented **versions** and view-oriented **versions** of engineering chang
e information in which the
engineering change information progresses through a set of status conditions and access to the **data** by **different** user groups is conditioned upon the status of the information. **Version** control software logic enables users to create **versioned** objects by logical key grouping of **data** elements. The **version** control logic acts upon the logical keys and speci
al **versioned**
attributes of these objects for the proper specification and selection of object instances during creation, **update** or retrieval processing. Insert and extract **sequence** numbers are automatically generated for both historical preservation of previous engineering change information and efficient retrieval of the currently effective designs. Instance level security facilitates the merger of **different** **versions** of the engineering change **data** having **different** engineering change status levels so that similar types of **data** can be contained within a single **data** base table.
:3

3. 5,008,748, Apr. 16, 1991, Signal coding; Michael D. Carr, et al., 348/417 [IMAGE AVAILABLE]
:end

=> s l5 and l6
L11 2 L5 AND L6

=> d bro
:1

1. 5,438,661, Aug. 1, 1995, **Version** **management** method and apparatus in multi-window environment; Tomoya Ogawa, 395/346, 356, 772 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,438,661 [IMAGE AVAILABLE] L11: 1 of 2

ABSTRACT:
The contents of a plurality of versions of a data file are displayed in respective windows on a display screen. Identifiers of parent windows, from which the windows are derived, and differential operation histories of the windows are stored in an operation history management table. The contents of the operation history management table are automatically renewed when a new window for a new version is generated and when the version of the data file is edited in a corresponding window. With this construction, comparison as

well as reference between versions on the same display screen is possible.
:2

2. 5,410,697, Apr. 25, 1995, Concurrency **management** using **version** identification of shared data as a supplement to use of locks; Robert Baird, et al., 395/479; 364/228.1, 243.41, 243.44, 246.8, 281.4, DIG.1; 395/726 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,410,697 [IMAGE AVAILABLE] L11: 2 of 2

ABSTRACT:
A method for managing concurrency using a serializing token as a supplement to locks for accessing the same page by different processes and ensuring coherence between data caches and a shared access backing store supporting the processes defined onto multiple processors. A shared lock and a new token are issued by a local lock manager (LLM) and local cache manager (LCM) in response to a page read request from a process. An exclusive lock, an invalidation of all cache resident copies of the page, a cache write through to backing store of the changed page, a copyback of a new token, and lock release are responsive to a page write/update request from a process where the token issued to the process during a prior request matches the token stored with a cache resident copy of the page. Otherwise, a write request fails and the process must first issue a read request to the page of interest.
:end

```
=> s (merg? or combin? or synchroniz?)/ab
          2918 MERG?/AB
        100452 COMBIN?/AB
         14823 SYNCHRONIZ?/AB
L12     116951 (MERG? OR COMBIN? OR SYNCHRONIZ?)/AB

=> d his

     (FILE 'USPAT' ENTERED AT 10:02:55 ON 16 MAY 1997)

                SET PAGE SCROLL
L1            4 S DELTA STREAM#
L2            1 S 5479654/PN
L3            0 S 5479654/UREF
L4         5348 S ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (
L5        27268 S ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (
L6           20 S (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/TI
L7          127 S (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/AB
L8          320 S (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/CLM
L9           72 S L5 AND L8
L10           3 S L4 AND L7
L11           2 S L5 AND L6
L12      116951 S (MERG? OR COMBIN? OR SYNCHRONIZ?)/AB

=> s (merg? or combin? or synchroniz?)/clm
         10044 MERG?/CLM
        319502 COMBIN?/CLM
         30991 SYNCHRONIZ?/CLM
L13     351332 (MERG? OR COMBIN? OR SYNCHRONIZ?)/CLM

=> s l9 and l13
L14          35 L9 AND L13
```

=> s l7 and l12
L15 9 L7 AND L12

=> s l14 and l7
L16 4 L14 AND L7

=> d bro
:1

1. 5,469,431, Nov. 21, 1995, Method of and apparatus for channel mapping with relative service identification; James W. Wendorf, et al., 370/254, 431, 478; 455/4.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,469,431 [IMAGE AVAILABLE] L16: 1 of 4

ABSTRACT:
The availability and location of a multiplicity of multiplexed digital information streams, transmitted over one of a multiplicity of channels, are identified by transmitting a Global Channel Map which includes a **version** number, one Band Map Table, one Channel Map Table, and at least as many Service Map tables as there are combinations of identical service types transmitted over different channels. The Band Map Table preferably contains an entry for each frequency band used by the system, and a Band Map Table **version** number. A Global Channel Map **Updates** file is transmitted in each band. Where a plurality of channels contain similar-types of services, one Service Map Table contains relative service identification numbers for those service types. The entries for relevant channels in the Channel Map Table contain base service identification data which are combined with the relative service identification number to locate a particular service transmitted over that channel.
:2

2. 5,327,556, Jul. 5, 1994, Fast intersystem page transfer in a data sharing environment with record locking; Chandrasekaran Mohan, et al., 395/608; 364/244.3, 246.8, 282.1, DIG.1; 395/610, 619 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,327,556 [IMAGE AVAILABLE] L16: 2 of 4

ABSTRACT:
A fast technique for transferring units of data between transaction systems in a shared disk environment. The owning system, having **updated** the page, generates a **version** number for the page which is stored with a lock possessed by the owning system. When a requesting system seeks a record on the page, its request for a lock illicit an indication that a more recent **version** of the page is required in the local memory. The buffer **management** component of a DBMS, with assistance from the lock **management**, triggers a memory to memory transfer of the page from the owning DBMS to the requesting DBMS using a low overhead communication protocol. The transfer of page is without disk I/O or the log I/O for the **updates** made to the page.
:3

3. 4,853,843, Aug. 1, 1989, System for merging virtual partitions of a distributed database; Denise J. Ecklund, 395/619; 364/222.81, 222.82, 222.9, 232.1, 240.8, 242.94, 242.95, 246, 246.3, 246.6, 246.8, 246.9, 256.3, 265, 265.1, 282.1, 282.3, 282.4, 283.2, 284.1, 284.3, DIG.1; 395/618 [IMAGE AVAILABLE]
:ab

US PAT NO: 4,853,843 [IMAGE AVAILABLE] L16: 3 of 4

ABSTRACT:
An object-oriented, distributed data base system separates into a plurality of virtual partitions following communication failure between sites accessing the data base. Each partition accesses a separate copy of an initial data base and independently **updates** groups of data objects included in the data base to add new **versions** of data objects to the data base. Each virtual partition maintains a copy of all previous **versions** of data objects and maintains a change list describing all group **updates** that it executes. Following restoration of communication between sites, each virtual partition merges the data bases maintained by separate partitions to form a consistent merged data base permitting **versions** of data objects and collections of data objects created by any one of the separate virtual partitions to be identified and accessed in the merged data base.
:4

4. 4,761,785, Aug. 2, 1988, Parity spreading to enhance storage access; Brian E. Clark, et al., 371/51.1, 2.2, 38.1, 40.2; 395/182.04 [IMAGE AVAILABLE]
:ab

US PAT NO: 4,761,785 [IMAGE AVAILABLE] L16: 4 of 4

ABSTRACT:
A storage **management** mechanism distributes parity blocks corresponding to multiple data blocks substantially equally among a set of storage devices. N storage units in a set are divided into a multiple of equally sized address blocks, each containing a plurality of records. Blocks from each storage unit having the same address ranges form a stripe of blocks. Each stripe has a block on one storage device containing parity for the remaining blocks of the stripe. Further stripes also have parity blocks, which are distributed on different storage units. Parity **updating** activity associated with every change to a data record is therefore distributed over the different storage units, enhancing access characteristics of the set of storage devices. The parity **updating** activity also includes the use of an independent **version** number stored with each data record and corresponding **version** numbers stored with the parity record. Each time a data record is changed, its **version** number is incremented and the corresponding **version** number in the parity record is incremented with the parity record **update**.
:end

```
=> s regress? (3a) test?
             9702 REGRESS?
           493428 TEST?
L17           278 REGRESS? (3A) TEST?

=> s l17/ab
              358 REGRESS?/AB
            33139 TEST?/AB
L18             8 (REGRESS?/AB (3A) TEST?/AB)

=> d bro
:1
```

1. 5,412,561, May 2, 1995, Method of analysis of serial visual fields; Joseph S. Rosenshein, et al., 351/224; 128/745; 351/226, 246; 382/128; 606/4; 607/91, 93 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,412,561 [IMAGE AVAILABLE] L18: 1 of 8

ABSTRACT:
A method of analyzing serial automated visual field data calculates a total field vector of the threshold sensitivity of points in a visual field and compares the total field vector with a calculated normal field vector for a given patient. The vector angle between the total field vector and the normal field vector is calculated to determine the distribution of threshold sensitivities to characterize the localization and the magnitude of field changes. In one embodiment, the sensitivities of each point in the visual field with respect to normal sensitivities for each point is plotted for each of a series of time spaced visual field tests. The single plot uses a line symmetrical about a zero axis whose magnitude reflects the difference from the norm for each point in each exam and forms a graphic depiction of changes in the visual field at each point over a series of visual field **tests**.
Linear **regression** analysis is p
erformed on the visual field data and
provides an indication of trends of visual field changes.
:2

2. 5,325,377, Jun. 28, 1994, Visual display signal processing system and method; Myron R. Tuttle, 371/67.1, 22.4 [IMAGE AVAILABLE]
:3

3. 5,218,605, Jun. 8, 1993, Software modules for testing computer hardware and software; Danny Low, et al., 395/183.21; 371/25.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,218,605 [IMAGE AVAILABLE] L18: 3 of 8

ABSTRACT:
A computer software-related device and method uses **regression** **testing** techniques for **testing** computer hardware and/or s
oftware application(s).
Input data and commands from a user are stored, and are sent to a hardware/software system under test. Signatures (representative of visual display data) which are received (with a selected prevalence) as a result of the sent input data and commands are also stored. On command of a user, the stored signatures, input data and commands are subsequently sent to the hardware/software system under test, and new signatures are generated. These new signatures are compared with the stored signatures, and the results of this comparison are used as an indication that the hardware/software system under test is performing as expected.
:4

4. 5,164,912, Nov. 17, 1992, Expert system tester; Robert L. Osborne, et al., 364/580, 551.01; 395/50, 75 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,164,912 [IMAGE AVAILABLE] L18: 4 of 8

ABSTRACT:
The present invention is an expert system test system which has the capability of exercising all combinations of data input types using specialized data test sets. The specialized data test sets include a set of sensor values for each level of an operating system. All levels of diagnosis can be tested without testing every possible sensor value and combination of sensor values. The test sets can be combined to produce various orders of testing allowing complex relationships between sensors and rules to be tested. The system includes a test plan which is used by the testing system

to iteratively apply the normal and specialized test data to the expert system. Changing the sensor values produces outputs for each iteration which are compared to expected results or to a baseline. Each test produces a log file which can be examined by the knowledge engineer. If exceptions are produced, the system produces an exception report which allows the knowledge engineer to determine whether the exception is abnormal or unexpected. An output log is also stored as historical data and used with future tests to obtain a complete **test regression** comparison using conventional database comparison tools. The system includes a test language which allows the user to formulate a test plan without the need for complicated programming associated therewith by designating the rulebase to be used, and by defining the order of the test and the tests performed therein, along with exceptions which include exceptions which include simple, set and sufficiency type exceptions.
:5

5. 5,157,782, Oct. 20, 1992, System and method for testing computer hardware and software; Myron R. Tuttle, et al., 395/183.21; 371/22.4, 25.1; 395/183.01, 183.22 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,157,782 [IMAGE AVAILABLE] L18: 5 of 8

ABSTRACT:
A computer related system and method uses **regression testing** techniques for **testing** computer hardware and/or software a
pplication(s). Input data
and commands from a user are stored on a host, and sent to an interface device, which then sends them to a hardware/software system under test. Visual display data on a visual display device is thereby affected. Signatures (representative of the visual display data) are generated by the interface device as a result of input data and commands sent to the interface device and system under test. These signatures are received and stored by the host. On command of a user, the stored signatures, input data and commands are subsequently sent to the hardware/software system under test, and new signatures are generated. These new signatures are compared with the stored signatures, and the results of this comparison are used as an indication that the hardware/software system under test is performing as expected.
:6

6. 5,153,886, Oct. 6, 1992, Visual display signal processing system and method; Myron R. Tuttle, 371/67.1, 22.4 [IMAGE AVAILABLE]
:7

7. 5,144,565, Sep. 1, 1992, Measurement of metallurgical properties in ferromagnetic test parts; Gordon R. Brown, et al., 364/507 [IMAGE AVAILABLE]
:8

8. 5,022,028, Jun. 4, 1991, Software verification apparatus; Edward J. Edmonds, et al., 395/183.14; 371/25.1; 395/183.08 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,022,028 [IMAGE AVAILABLE] L18: 8 of 8

ABSTRACT:
An apparatus for verification of target software loaded in a target computer. The apparatus includes communication and monitoring circuits inserted in the target computer and in a host computer which is programmed to direct operation of target software tests through the communication and monitoring circuits. The communication and monitoring circuits transmit external stimuli

to the target software in a non-intrusive manner by, for example, keyboard simulation. The communication and monitoring circuits also capture target software output data for comparison with reference data stored in the host computer. Both of these operations are carried out in a non-intrusive manner in hardware, (the communication and monitoring circuits). The external stimuli including test instructions may be either recorded in a manual recording session or may be generated in the host computer. Similarly, the target software output reference data may be either generated in the host computer or recorded in a manual session. The apparatus is particularly suitable for **regression testing** and enables a user to detect faults at any stage of software development irrespective of how many updates are involved.
:end

=> d l15 bro
:1

1. 5,550,685, Aug. 27, 1996, Applying an adaptive feed-forward algorithm as a frequency selective filter in a closed loop disk drive servo system in order to compensate for periodic perturbations which otherwise appear in the servo system position error signal; David M. Drouin, 360/77.08, 77.04 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,550,685 [IMAGE AVAILABLE] L15: 1 of 9

ABSTRACT:
A method for applying an adaptive feed-forward waveform correction to a primary servo loop compensation signal in a rotating data storage apparatus. A position error signal is used to determine a fixed feed-forward correction upon initialization or other predetermined conditions. An adaptive feed-forward waveform correction is periodically determined by adding a scaled **version** of the position error signal to the stored adaptive feed-forward value, and then processing the resulting sum through a frequency selective filter. The stored adaptive feed-forward correction is then **updated** with this new result. Simultaneously, the **updated** adaptive feed-forward correction is **combined** with the fixed feed-forward correction and the position error signal to generate a primary servo loop compensation signal for controlling the read/write head of the data storage apparatus.
:2

2. 5,546,579, Aug. 13, 1996, Page refreshing procedure using two locking granularities to ensure cache coherency in a multisystem database processing environment having a high-speed shared electronic store; Jeffrey W. Josten, et al., 395/608; 364/134, 243.41, 935.43, DIG.1, DIG.2; 395/183.14 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,546,579 [IMAGE AVAILABLE] L15: 2 of 9

ABSTRACT:
A method for ensuring data coherence while detecting whether the locally cached copy of a data page is invalid and responsively refreshing the locally cached page from a Shared Electronic Store (SES) in a multisystem shared disk environment. Locally cached data pages may become invalid in a multisystem shared disk environment because of transactions executed by other systems for the common database. Thus, whenever a transaction in a database **management** system (DBMS) instance desires to read or **update** a record in a locally cached data page, the DBMS must first verify validity for the locally cached copy and, for stale or invalid copies, must re-read and re-register the

latest **version** in the SES. This invention provides a procedure for the necessary verification and refreshing steps that relies on page latching to serialize the **combination** of local multiuser and global multisystem activities. The procedure of this invention supports both record and page locking granularities.
:3

3. 5,469,431, Nov. 21, 1995, Method of and apparatus for channel mapping with relative service identification; James W. Wendorf, et al., 370/254, 431, 478; 455/4.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,469,431 [IMAGE AVAILABLE] L15: 3 of 9

ABSTRACT:
The availability and location of a multiplicity of multiplexed digital information streams, transmitted over one of a multiplicity of channels, are identified by transmitting a Global Channel Map which includes a **version** number, one Band Map Table, one Channel Map Table, and at least as many Service Map tables as there are **combinations** of identical service types transmitted over different channels. The Band Map Table preferably contains an entry for each frequency band used by the system, and a Band Map Table **version** number. A Global Channel Map **Updates** file is transmitted in each band. Where a plurality of channels contain similar-types of services, one Service Map Table contains relative service identification numbers for those service types. The entries for relevant channels in the Channel Map Table contain base service identification data which are **combined** with the relative service identification number to locate a particular service transmitted over that channel.
:4

4. 5,367,306, Nov. 22, 1994, GPS integrated ELT system; Blake D. Hollon, et al., 342/386, 357 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,367,306 [IMAGE AVAILABLE] L15: 4 of 9

ABSTRACT:
Method and apparatus are described for the integration of global positioning system (GPS) or LORAN positioning information into the transmission of an emergency locator transmitter (ELT) radio beacon signal. Signals from NAVSTAR satellites or LORAN system are received by a receiver. The calculated position, and then subsequent **updates**, are processed and alternately stored in two locations on a static random access memory chip. Upon activation of the ELT transmitter, the **updated** position information, or the latest stored information, is **combined** with aircraft information stored on an EPROM chip. The aircraft information and position information is preferably converted from a digital-to-speech signal, by a voice synthesizer. All this information is then integrated into the radio beacon signal and broadcast by the ELT. This new ELT-GPS storing and processing package is self-contained, independently powered, and crash resistant. This system avoids loss of time in establishing the location of the beacon signal. A ground use **version** of the system is also described.
:5

5. 5,317,729, May 31, 1994, Method for the storage of multi-versioned data with retrieval based on searched query; Sujan K. Mukherjee, et al., 395/603; 364/282.1, 283.1, DIG.1; 395/619 [IMAGE AVAILABLE]
:6

6. 5,131,012, Jul. 14, 1992, Synchronization for cylic redundancy check based, broadband communications network; Subrahmanyam Dravida, 375/357; 371/42, 47.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,131,012 [IMAGE AVAILABLE] L15: 6 of 9

ABSTRACT:
A received digital signal is **synchronized** with a receiver in a communications network by employing a code word which is **updated** in accordance with a unique **update** technique. The **synchronization** arrangement receives the digital signal at a receiver, generates an error check code word over a prescribed portion of the received digital signal and compares the generated error check code word with an expected error check code word in the received digital signal. In response to the comparison result, the generation of the code word and the comparison is iteratged until **synchronization** is obtained. A new code word is generated as a function of a modified **version** of the last previously generated code word (previous code word), a function term based on a set of bits dropped from the prescribed portion of the received digital signal, and a set of bits added to the prescribed portion of the received digital signal. In one embodiment, a cyclic redundancy check (CRC) code word is employed. In one instance, the modified **version** of the code word is obtained by shifting the previous code word to the left dependent on the lead set of bits in the previous code word. In another instance, the modified **version** of the code word is obtained by shifting the previous code word to the left and then adding the CRC generator polynomial to it, also dependent on the lead set of bits in the previous code word. The function term is representative of a remainder obtained by modifying the dropped set of bits in a predetermined manner and by dividing the modified dropped set of bits by a prescribed CRC generator polynomial. The new CRC code word is then obtained by adding the modified **version** of the previous code word, the function term and the set of added bits.
:7

7. 4,853,843, Aug. 1, 1989, System for merging virtual partitions of a distributed database; Denise J. Ecklund, 395/619; 364/222.81, 222.82, 222.9, 232.1, 240.8, 242.94, 242.95, 246, 246.3, 246.6, 246.8, 246.9, 256.3, 265, 265.1, 282.1, 282.3, 282.4, 283.2, 284.1, 284.3, DIG.1; 395/618 [IMAGE AVAILABLE]
:8

8. 4,675,886, Jun. 23, 1987, Frame synchronization device; Serge Surie, 375/368; 327/155, 160; 375/371 [IMAGE AVAILABLE]
:ab

US PAT NO: 4,675,886 [IMAGE AVAILABLE] L15: 8 of 9

ABSTRACT:
Frame **synchronization** devices utilize a frame alignment word decoder connected to the outputs of a shift register which receives on its input the received data bit stream. It is clocked by a clock signal generated from a selection of periods of the data bit stream timing signal reproducing a periodic pattern. This pattern is formed by relative bit locations within the duration of a frame certain at least of which are distributed according to the distribution of the bits of an alignment word in a frame and which form groups of the same size regularly distributed over the duration of a frame. This clock signal is generated in the device by a divider which divides by 20 or by 21 which imposes on it a periodic phase skip by the value of one period of the data bit stream timing for as long as the alignment word is not recognized by the decoder. The shift register is implemented in two parallel

parts clocked by **versions** of the clock signal with a relative phase shift between them, one of which parts **updates** the other part in parallel on each phase skip of the clock signal.
:9

9. 3,997,718, Dec. 14, 1976, Premium interactive communication system; Luther W. Ricketts, et al., 348/13; 380/13, 20; 455/2, 5.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 3,997,718 [IMAGE AVAILABLE] L15: 9 of 9

ABSTRACT:
A cable television and communication system is disclosed which is suitable for community antenna television (CATV) closed circuit television (CCTV) and other types of signal distribution systems with service function applications such as for use in hotel, motel, apartment complexes, and the like. The system has the capability of distribution and subscriber reception of unencoded and encoded or limited access video and audio programs with simultaneous two way digital data communication. The subscriber units are interconnected by a tree-organized wideband communication link such as co-axial cable system with a network central unit. Subscriber unit identification control and data exchange is accomplished by the use of a high speed time-slot organized format with each subscriber being assigned a predetermined unique television **synchronization** related time-slot. The central unit utilizes a small digital computer which functions to provide network supervision and **management** of subscriber requests, accounting, billing and other processing such as viewing analysis. The system utilizes a modular configuration which allows for a low cost and simple initial installation capable of being later expanded to a more sophisticated **version**.
:end

=> s 5317729/uref
L19 9 5317729/UREF

=> d bro
:1

1. 5,600,832, Feb. 4, 1997, Variant domains and variant maps in a versioned database management system; Neal R. Eisenberg, et al., 395/619; 364/282.3, 283.1, DIG.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,600,832 [IMAGE AVAILABLE] L19: 1 of 9

ABSTRACT:
A method is provided in a versioned database management system for allowing parts to be versioned according to different variant hierarchies, and for maintaining correct configurations of versions of parts as those parts are drawn down, changed, and promoted. Each version of a part is associated with a variant domain. Each variant domain is represented by a single variant hierarchy whose levels can be used to reference versions of parts in that variant domain and controls how versions of parts in that variant domain are drawn down, changed, and promoted. Variant domain default groups are provided so that tools may add new parts without specifying the variant domains to which the parts are to belong. In order to control which versions of parts and from which version domains are visible, variant maps are defined by the user installation. When a change is made for one configuration, it is simultaneously made in any other configuration identified by a variant map which includes the same variant ID. A variant map thus constructed is used by the VDMS to determine the variant hierarchy level to which any instances

created belong, and to determine which instances in a promote group are promoted and to which hierarchy level they are promoted.
:2

2. 5,592,661, Jan. 7, 1997, Detection of independent changes via change identifiers in a versioned database management system; Neal R. Eisenberg, et al., 395/615; 364/282.1, 282.3, 283.1, DIG.1 [IMAGE AVAILABLE]
:xa

US PAT NO: 5,592,661 [IMAGE AVAILABLE] L19: 2 of 9
ASST-EXMR: Jean R. Homere
:3

3. 5,581,755, Dec. 3, 1996, Method for maintaining a history of system data and processes for an enterprise; Paul D. Koerber, et al., 395/614; 364/268, 964.34, DIG.1, DIG.2; 395/670, 701 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,581,755 [IMAGE AVAILABLE] L19: 3 of 9

ABSTRACT:
The method of the present invention is useful in a computer system having a user interface, a memory, a repository and a database. The method is a repository program executed by the computer system for maintaining a history of objects stored in the repository. The method comprises the steps of determining if the object is a new logical object, and if so constructing a new versioned object; and, if not retrieving from the repository an object on which a new state is to be based. If the new state is being derived from only one object, then reserving the object. If the state is being derived from two objects, then retrieving from the repository the other object on which the new state is based, and merging the two objects. If the state changes are to be retained, then replacing the object; and, if the state changes are not to be retained, then unreserving the object.
:4

4. 5,539,906, Jul. 23, 1996, Method and apparatus for controlling access to data elements in a data processing system based on status of an industrial process; Robert L. Abraham, et al., 395/609; 364/246.6, 246.8, DIG.1 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,539,906 [IMAGE AVAILABLE] L19: 4 of 9

ABSTRACT:
The security of data elements which represent an industrial process, which are manipulated by users on a data processing system and in which the industrial process includes a series of industrial process steps, are controlled by permitting groups of users to access predetermined data elements based on the industrial process step at which the industrial process is currently active. A user is prevented from accessing the requested element if the industrial process is not at an industrial process step corresponding to one of the industrial process steps for which the user has authority to access the data element. Thus, access to data is prevented based on the status of the data, in addition to the type of data. When selected database elements are associated with one of many locations, access is also denied to a user based on the location. Security access based on status and location may be provided in response to a change in the current industrial process step. Access authority to the data elements is changed compared to the access authority at the immediately preceding industrial process step based on mappings in one or more tables. Improved security of data elements which

represent an industrial process is thereby provided.
:5

5. 5,504,879, Apr. 2, 1996, Resolution of relationship source and target in a versioned database management system; Neal R. Eisenberg, et al., 395/611; 364/282.1, 282.3, DIG.1; 395/619 [IMAGE AVAILABLE]
:xa

US PAT NO: 5,504,879 [IMAGE AVAILABLE] L19: 5 of 9
ASST-EXMR: Wayne Amsbury
:ab

US PAT NO: 5,504,879 [IMAGE AVAILABLE] L19: 5 of 9

ABSTRACT:
A versioned data management system is provided with a method for resolving sources and targets of relationships. For each entity instance, a lifetime ID is recorded. When the add interface is used to add an entity, a value is assigned to the lifetime ID, which value has never been used before for an instance of the entity type. When the update interface is used to update an entity, the lifetime ID is maintained unchanged. If the update results in a new version, the new version is given the same lifetime ID value as that for the version that was the basis for the update. If the delete interface is then used to delete the entity, and the add interface is later used to add an instance with the same part key, that instance will have a different lifetime ID. For each relationship instance, lifetime IDs are recorded for the relationship, its source, and its target. When the add interface is used to add a relationship, a value is assigned to the lifetime ID, which value has never been used before for an instance of the relationship type. Also, the lifetime IDs of the source and target of the relationship are recorded in the relationship instance. The lifetime ID of the relationship and the lifetime IDs of the source and target are maintained unchanged. If an update results in a new version of the relationship, the new version is given the same lifetime ID value as that for the version that was the basis for the update.
:6

6. 5,434,791, Jul. 18, 1995, Product structure management; Boma R. Koko, et al., 364/468.03, 468.14; 395/614 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,434,791 [IMAGE AVAILABLE] L19: 6 of 9

ABSTRACT:
An object-oriented method of using a computer to store a model of an imprecise structure of a product. The product's components are modeled as items and item revisions. Each item and item revision has a view, which may have view revisions. The method links view objects and view revision objects with occurrence references to each other and to view objects and view revision objects of other components. Context-specific view revisions are modeled as appearances. A user's request for a display of a product is received and used to invoke configuration rules that determine which view revisions) are part of the product. The correct view revisions are assembled with their occurrences and appearances.
:7

7. 5,428,795, Jun. 27, 1995, Method of and apparatus for providing automatic security control of distributions within a data processing system; William J. Johnson, et al., 395/728; 340/825.31, 825.34; 364/222.5, 222.6, 264, 264.5, 286.4, 286.5, DIG.1; 380/4, 25; 395/609 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,428,795 [IMAGE AVAILABLE] L19: 7 of 9

ABSTRACT:
Automatic detection of security violation objects within distributions and appropriate handling of those violations is provided. A confidential object dictionary contains definitions of objects which violate security. Distributions are automatically searched for objects contained in the dictionary. A violation log may be used for summarizing violations and determining an appropriate predefined action for a violation. The appropriate predefined action for a violation may include third party notification, notifying third parties in priority order, safe storing the violation, or inhibiting distribution of the violation.
:8

8. 5,428,774, Jun. 27, 1995, System of updating an index file of frame sequences so that it indexes non-overlapping motion image frame sequences; Junichi Takahashi, et al., 395/612; 364/282.1, 282.3, DIG.1; 386/68; 395/328, 615, 807, 960 [IMAGE AVAILABLE]
:ab

US PAT NO: 5,428,774 [IMAGE AVAILABLE] L19: 8 of 9

ABSTRACT:
In the creation of the index file, one index record is allocated to each set consisting of one retrieval key and the initial and final positions in the motion image of one frame to which the retrieval key is allocated, and information on this set is stored therein. In the query processing, for a query specified by a Boolean expression of retrieval keys, records whose keywords match any of the retrieval keys included in the query are retrieved from the index file. Thereafter, all the initial and final positions of the matching records are arranged along a time axis. For each of intervals defined by those positions, a truth table is created in which the allocation of the retrieval keys included in the query is expressed by truth values, and the logical condition of the query is evaluated based on the table.
:9

9. 5,386,559, Jan. 31, 1995, Variant domains and variant maps in a versioned database management system; Neal R. Eisenberg, et al., 395/617; 364/282.1, 282.3, 283.1, DIG.1 [IMAGE AVAILABLE]
:xa

US PAT NO: 5,386,559 [IMAGE AVAILABLE] L19: 9 of 9
ASST-EXMR: John C. Loomis
:ab

US PAT NO: 5,386,559 [IMAGE AVAILABLE] L19: 9 of 9

ABSTRACT:
A method is provided in a versioned database management system for allowing parts to be versioned according to different variant hierarchies, and for maintaining correct configurations of versions of parts as those parts are drawn down, changed, and promoted. Each version of a part is associated with a variant domain. Each variant domain is represented by a single variant hierarchy whose levels can be used to reference versions of parts in that variant domain and controls how versions of parts in that variant domain are drawn down, changed, and promoted. Variant domain default groups are provided so that tools may add new parts without specifying the variant domains to which the parts are to belong. In order to control which versions of parts and from which version domains are visible, variant maps are defined by the

user installation. When a change is made for one configuration, it is simultaneously made in any other configuration identified by a variant map which includes the same variant ID. A variant map thus constructed is used by the VDMS to determine the variant hierarchy level to which any instances created belong, and to determine which instances in a promote group are promoted and to which hierarchy level they are promoted.
:end

=> d his full

(FILE 'USPAT' ENTERED AT 10:02:55 ON 16 MAY 1997)

SET PAGE SCROLL
L1 4 SEA DELTA STREAM#
L2 1 SEA 5479654/PN
L3 0 SEA 5479654/UREF
L4 5348 SEA ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (CHARACTER# OR BYTE OR DATA OR BIT#) AND (MATCH? OR COMPAR? OR DIFFER? OR DELTA))/AB
L5 27268 SEA ((SEQUEN? OR STREAM# OR CHAIN# OR SEGMENT# OR FRAME#) AND (CHARACTER# OR BYTE OR DATA OR BIT#) AND (MATCH? OR COMPAR? OR DIFFER? OR DELTA))/CLM
L6 20 SEA (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/TI
L7 127 SEA (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/AB
L8 320 SEA (VERSION? AND (REVIS? OR UPDAT? OR MANAG?))/CLM } and (differ? or delta
L9 72 SEA L5 AND L8
L10 3 SEA L4 AND L7
L11 2 SEA L5 AND L6
L12 116951 SEA (MERG? OR COMBIN? OR SYNCHRONIZ?)/AB
L13 351332 SEA (MERG? OR COMBIN? OR SYNCHRONIZ?)/CLM
L14 35 SEA L9 AND L13
L15 9 SEA L7 AND L12
L16 4 SEA L14 AND L7
L17 278 SEA REGRESS? (3A) TEST?
L18 8 SEA (REGRESS?/AB (3A) TEST?/AB)
L19 9 SEA 5317729/UREF

FILE USPAT

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*                W E L C O M E   T O   T H E                    *
*          U. S.   P A T E N T   T E X T   F I L E              *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

=>

User ID:	d237jrh
User Name:	JEAN R. HOMERE
User Phone:	0003086647
Workstation Id:	WSHBUEBZ
Printer Id:	webzptr
Job #	0019